VTT Technical Research Centre of Finland

Automatic Cloud Detection Method Based on Generative Adversarial Networks in Remote Sensing Images

Li, Jun; Wu, Zhaocong; Hu, Zhongwen; Zhang, Yi; Molinier, Matthieu

Published in:
ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences

DOI:
10.5194/isprs-annals-V-2-2020-885-2020

Published: 03/08/2020

Document Version
Publisher's final version

AUTOMATIC CLOUD DETECTION METHOD BASED ON GENERATIVE ADVERSARIAL NETWORKS IN REMOTE SENSING IMAGES

Jun Li [1, *], Zhaocong Wu [1], Zhongwen Hu [2], Yi Zhang[1] and Matthieu Molinier [3]

[1] School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, China - (jun__li, zcwoo, ivory2008)@whu.edu.cn
[2] College of Life Sciences and Oceanography, Shenzhen University, Shenzhen, China - zwhoo@szu.edu.cn
[3] VTT Technical Research Centre of Finland Ltd, Espoo, Finland - matthieu.molinier@vtt.fi

Commission TC II, WG II/III

KEY WORDS: Cloud detection, Generative adversarial networks (GANs), Attention mechanism, Deep learning, Auto-GAN.

ABSTRACT:

Clouds in optical remote sensing images seriously affect the visibility of background pixels and greatly reduce the availability of images. It is necessary to detect clouds before processing images. In this paper, a novel cloud detection method based on attentive generative adversarial network (Auto-GAN) is proposed for cloud detection. Our main idea is to inject visual attention into the domain transformation to detect clouds automatically. First, we use a discriminator (D) to distinguish between cloudy and cloud free images. Then, a segmentation network is used to detect the difference between cloudy and cloud-free images (i.e. clouds). Last, a generator (G) is used to fill in the different regions in cloud image in order to confuse the discriminator. Auto-GAN only requires images and their labels (1 for a cloud-free image, 0 for a cloudy image) in the training phase which is more time-saving to acquire than existing methods based on CNNs that require pixel-level labels. Auto-GAN is applied to cloud detection in Sentinel-2A Level 1C imagery. The results indicate that Auto-GAN method performs well in cloud detection over different land surfaces.

1 INTRODUCTION

Remote sensing images have been applied into many fields such as change detection, land cover and land use classification and environmental monitoring (Novo-Fernández et al., 2018). However, 66% of the Earth's surface is covered by clouds most of the time (Zhang et al, 2004). This blocks the signal from land surface and alters the reflectance of ground objects, reducing the applicability of optical images (Fisher, 2014). Since the transmittance of thick clouds in optical images is 0, the signals of ground objects are completely blocked and highlights such as bare land and buildings are easily confused with them. Clouds not only block the ground objects, but also affect the subsequent processing of image fusion, registration. Although humans can very accurately label cloud masks, this process is very time-consuming, expensive and difficult. Thus, the automatic cloud detection in optical remote sensing images is very important.

Over the past decades, many methods have been proposed for cloud detection. Generally, the traditional methods can be divided into two types: threshold-based methods and multitemporal-based methods. Threshold-based methods are widely used to generate basic masks, and can distinguish cloudy from clear-sky pixels employing the spectral differences between dark land and clouds (Sun et al., 2018; Wang et al., 2016; Zhu et al., 2015; Parmes et al., 2017) but often fail to separate clouds from highlights. Multitemporal-based methods use clean images from different time periods within a certain area to produce clean synthetic images, then calculate the difference between cloudy images and clean images.

Wei et al. (2016) select visible-to-NIR bands to separate land surfaces from clouds, and use the short-wave infrared bands to distinguish clouds from snow/ice in MODIS and Landsat 8 data. Zhu et al. (2012) propose Fmask, a rule-based automated cloud detection method for cloud detection in Landsat 8 images. Frantz et al. (2015) use the spectrally correlated NIR bands, which are additionally affected by a view angle parallax to separate clouds and land surfaces, and further improves the separation of potential cloud pixels (PCPs) produced by Fmask. Han et al. (2014) firstly detects thick clouds using a threshold method then uses a modified scale-invariant feature transform (SIFT) method to transform cloud-free reference images (acquired from the same region at a different time) to the coordinates of the cloudy image.

Some disadvantages of the traditional methods are: 1) threshold-based methods are based on human experience and professional knowledge. It is very difficult to set a threshold that can be used for other satellite images; 2) Method based on multi-threshold for multi-spectral bands is limited by the bands of satellite sensor (Parmes et al., 2017); 3) Multi-temporal methods require time series images which are not always available or practical to process. They may be suitable for the specific regions, but do not work well in other regions.

Recently, many methods based on machine learning and especially deep learning (DL) have been proposed for cloud detection in remote sensing. Mateo et al. (2017) propose a deep learning-based method for cloud detection in Proba-V multispectral images. A supervised CNN architecture for cloud detection in SPOT6 images is proposed in (Goff et al., 2017). Xie et al. (2017) firstly transform RGB to HIS color space, then clusters the image into super-pixels by a saliency detection method GS04 (Zhao et al., 2015), then train a CNNs model to detect clouds. U-Net architecture has been proved effective for on-board cloud detection in small satellite images (Zhang et al.

* Corresponding author

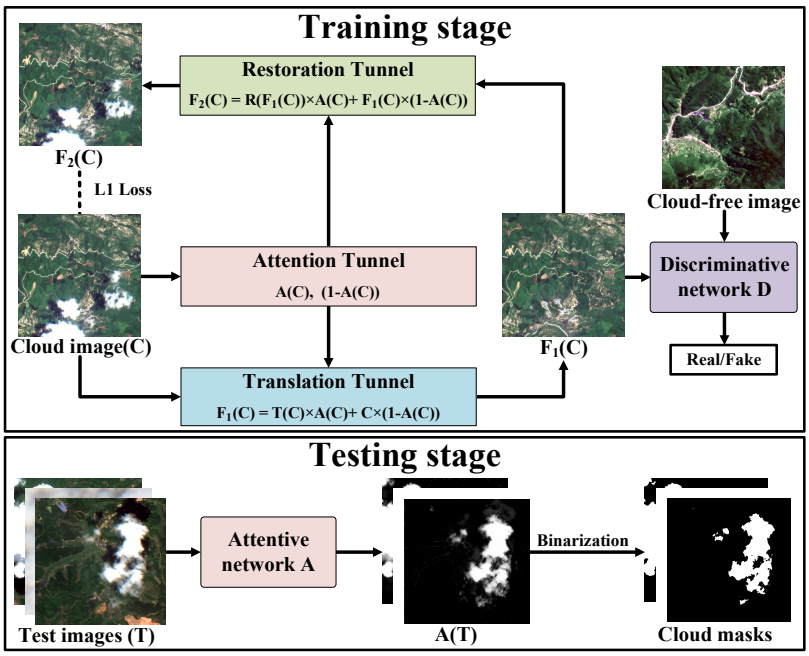

Figure 1. Flowchart of cloud detection based on Auto-GAN.

2018). Multi-scale convolutional features are used to detect cloud in medium and high-resolution remote sensing images of different sensor (Li et al. 2019). Although these DL-based methods have achieved very high accuracy for cloud detection in remote sensing images, they require pixel-level ground truths labeled by humans which are very time-consuming to obtain. Thus, an unsupervised feature extraction method that do not require pixel-level ground truths is more desirable.

Generative adversarial networks (GANs) have been proposed as an unsupervised deep learning model (Goodfellow et al., 2014). It is a generative architecture in which two networks play a minimax game: a generative network translates a random input into a realistic sample, and a discriminative network distinguishes the generated sample from the true sample. Isola et al. (2017) propose a pix2pix GANs framework for image-to-image translation with paired images; this method can realize the translation between an aerial photo and a map. Zhu et al. (2017) propose a method for image-to-image translation with unpaired images using cycle consistency loss to train G and D to be consistent with each other. Due to its effectiveness, GANs is one of the most promising methods for unsupervised learning on complex distributions.

In addition, the human attention mechanism has been widely studied in recent years. By introducing attention mechanism, the neural network can focus more attention on an area of interest (Vaswani et al., 2017). The method proposed in (Qian et al, 2018) combines visual attention and GANs to remove raindrop in images; it adopts a LSTM network to locate the raindrop regions of the input image which will guide the generative and discriminative networks to pay more attention to raindrop regions and ignore other regions without raindrop.

In this paper, a method based on GANs for automatic cloud detection is proposed, where the GANs architecture is redesigned and injected with attention mechanism to detect clouds. GANs is used to translate the detected regions between cloud and background and help attention concentrate detected regions on clouds. Experimental results on Sentinel-2A images in China show that the proposed method performs well under different background conditions both in vision and quality.

The rest of this paper is organized as follows. Section 2 introduces the framework of the proposed Auto-GAN method and the details. Section 3 describes the experimental data and setup, and discusses the comparative results with the official Sentinel-2 cloud masks (sen2cor) and baseline deep learning-based methods. Finally, the conclusions are drawn in Section 4.

2 METHODOLOGY

The inputs for the Auto-GAN method are an image and a single corresponding image-level label (whether there are clouds in the input image). Auto-GAN aims to detect cloud regions automatically by extracting the feature difference between unpaired cloudy and cloud-free images. The proposed method consists of four networks: an attentive network, two generative networks and a discriminative network. The discriminative network is used to distinguish whether there are clouds in the input image. The attentive network is used to detect and delineate cloud regions in the input image. One of the generative networks is used to translate the cloud regions into cloud-free regions. The other generative network is used to restore the translated cloud-free regions back into cloud regions.

2.1 Overview of the Proposed Method

GANs were originally designed to produce samples. The basic GAN architecture contains two networks: a generative network (G) and a discriminative network (D). G tries to generate samples in order to confuse D, while D tries to distinguish real samples from generated samples (Goodfellow et al, 2014). This game can be represented by the following formulation:

$$\min_G \max_D E_{t\sim d_t}[\log D(t)] + E_{s\sim d_s}[\log(1 - D(G(s)))] \quad (1)$$

where d_t = distribution of target samples
d_s = distribution of source samples

As shown in Figure 1, Auto-GAN consists of three tunnels (attention tunnel, translation tunnel and restoration tunnel) and a

discriminative network. The attention tunnel is used to detect cloud regions which is also called attention map in this work. And we use a segmentation network to produce the attention map of the cloud which will guide the translation and restoration processes to pay more attention on cloud regions. The attention map is represented by a matrix P = μ [0,1] of grayscale values ranging from 0 to 1, with higher values in P representing more the attention to the corresponding region. The translation tunnel aims to translate a cloudy image into a cloud-free image, which is then put into the discriminative network to discriminate whether it contains clouds or not. In order to confuse the discriminative network, the cloud regions need to be translated should be as large as possible. The restoration tunnel aims to restore the translated cloud-free image back to a cloudy image, which is then compared with the original input cloud image. By minimizing the difference of global consistency between the restored image and the original image, the translated/restored regions should be as small as possible. This trade-off between large translated regions and small restored regions can guide the attentive network to concentrate more attention on cloud regions automatically.

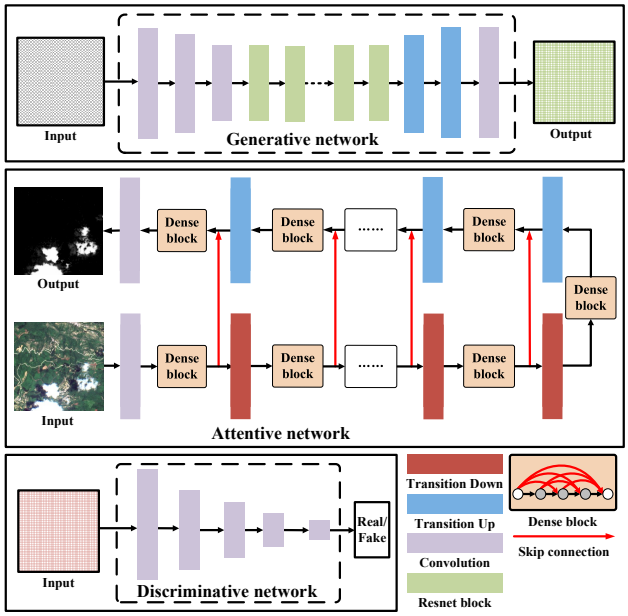

Figure 2. Detail components of generative network, attentive network and discriminative network.

The detail components of the generative networks, attentive network and discriminative network are further shown in Figure 2. Inspired by (Jégou et al, 2017) for image segmentation, the attentive network (A) in Auto-GAN adopts the FC-DenseNet architecture which combines the features of down-sampled output with the up-sampled output at the same spatial resolution level. A consists of seventy-three layers of operations, including convolution, transpose convolution, instance normalization, Rectified Linear Unit (ReLU), Leaky Rectified Linear Unit (LReLU) and sigmoid activation functions. The architecture of the two generative networks (T and R) are both based on ResNet architecture (He et al, 2016) which can prevent vanishing or exploding gradients and ensure the integrity of the input information. T and R have the same architecture which consists of twenty-five layers of operations, including convolution, transpose convolution, instance normalization, ReLU, LReLU and tanh. The discriminative network (D) is based on PatchGAN (Isola et al, 2017) and contains thirteen layers including convolution, instance normalization, and LReLU. The output of D is a 16×16 matrix which represents a large receptive field from the original image.

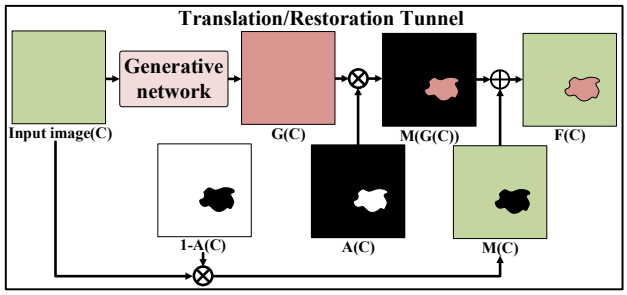

Figure 3. Detail operations of Translation/Restoration Tunnel.

2.2 Expansion with translation network

In this phase, we put a cloud image c into the attention network A to detect cloud regions and get a rough attention map A(c). T is utilized to translate the cloud regions into cloud-free regions (T(c)), which will be used to replace regions where cloud were detected in A(c). As shown in Figure 3, the input image and T(c) perform a mask operation with A(c). The mask operation is as follows:

$$F_1(c) = T(c) \times A(c) + c \times (1 - A(c)) \quad (2)$$

Where c = input cloud image
T(c) = generated background image by T
$F_1(c)$ = fused cloud-free image
1 = a matrix of 1 with the same size as A(c)

Then, the discriminative network (D) is used to assess the quality of the fused cloud-free image ($F_1(c)$) by measuring the feature difference between the fused cloud-free image ($F_1(c)$) and a real cloud-free image (x). The real cloud-free image (x) is not necessarily from the same location, but should contain similar land covers. We give the expression of the loss function of D as follows:

$$L_D = \frac{1}{2}\{E_{c\sim d_c}[D(F_1(c))^2] + E_{x\sim d_n}[(D(x) - 1)^2]\} \quad (3)$$

Where d_c = distribution of cloud images
d_n = distribution of cloud-free images

The loss function of F_1 which concerns T and A is:

$$L_{F_1} = E_{c\sim d_c}[(D(F_1(c) - 1))^2] \quad (4)$$

We adopt the least squares function as our loss function for D. If we would use the cross entropy as the loss function, the generator would not optimize the generated images that are recognized as real images by D even if these generated images are still far away from the decision boundary of D, which means that the image translated by F_1 would not be of high quality. The least squares method is different. In order to minimize the loss of the least squares function, on the premise of confusing D, the generative network T will pull the generated image closer to the decision boundary, where confusion is more likely.

We assume that T(c) is a high-quality cloud-free image which can confuse D. So, if $F_1(c)$ is to confuse D, A(c) produced by the attentive network A need to be as large as possible.

2.3 Reduction with restoration network

The adversarial loss alone can only guide A to expand detected regions A(c), and T to generate a high-quality background image.

In order to constrain A to focus only on cloud regions (which means that the translation tunnel only translates cloud regions and keeps background regions unchanged), another generative network R is used to restore the translated background into cloud ($R(F_1(c))$). As shown in Figure 3, the restoration process is also a mask operation. The restoration process is as follows:

$$F_2(c) = R(F_1(c))\times A(c)+F_1(c)\times(1-A(c))$$
$$= R(F_1(c))\times A(c)+[T(c)\times A(c)+c\times(1-A(c))]\times(1-A(c))$$
$$= [R(F_1(c))+T(c)]\times A(c)+c\times(1-A(c))^2 \quad (5)$$

It can be seen that the restored image $F_2(c)$ is fused with two components: $R(F_1(c)) + T(c)$ and c. The fusion factor is A(c). We compare $F_2(c)$ with the original input cloud image c to assess the effect of the restoration process as follows:

$$L_{F_2} = E_{c\sim d_c}[\|F_2(c)-c\|_1] \quad (6)$$

We adopt the absolute loss as the loss function of F_2 which involves A and R. We call this the global cycle consistency loss. The absolute loss can assess the absolute difference between $F_2(c)$ and c and can avoid image blur.

As shown in equation (5) and (6), to minimize L_{F_2}, A(c) should be as small as possible. For example, when $A(c) = 0$, $F_2(c) = c$, $L_{F_2} = 0$. Thus, the restoration tunnel can constrain A to reduce the detected regions and guide R to restore the translated background of $F_1(c)$ back to a cloud region.

The translation tunnel tries to expand the translated regions to produce high quality cloud-free images to confuse D, and the restoration tunnel tries to reduce the restored regions to keep as much information of original image as possible in order to make the restored image $F_2(c)$ and the original image c globally consistent. So, we train them together and the loss function of A, T and R can be combined into the Auto-GAN loss as follows:

$$L_{AG} = E_{c\sim d_c}[(D(F_1(c))-1)^2] + \lambda E_{c\sim d_c}[\|F_2(c)-c\|_1] \quad (7)$$

with λ a weight parameter between T and R loss functions. The value of L_{AG} is fed to A, T and R. In order to minimize L_{AG}, the networks A, T and R will optimize their parameters. After A, T and R being well-trained, A can detect clouds accurately, T can translate cloudy images into cloud-free images and R can restore the cloud-free images back into cloudy images.

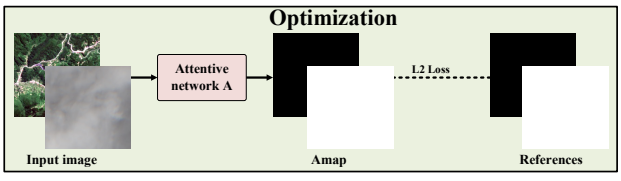

Figure 4. Optimization process of Attention network.

2.4 Optimization

In order to improve the detection accuracy of A, we consider both cloud-free images and images full of clouds. As shown in Figure 4, we put them into A to produce cloud attention maps. So, the proposed method introduces another algorithm for the optimization of A to make the best use of spectral information. According to the spectral information of input images, the attention maps of cloud-free and fully cloudy images should be matrices of 0 and 1, respectively. The optimization function is as follows:

$$L_A = \frac{1}{2}\{E_{f\sim d_f}[(A(f)-1)^2] + E_{x\sim d_n}[(A(x)-0)^2]\} \quad (8)$$

Where d_f = distribution of image full of cloud
0 = a matrix of 0 with the same size as A(x)

We adopt the least squares function that can speed up network convergence as the loss function of the optimization process. This loss function takes two extreme cases into consideration: cloud-free and full of cloud. To minimize L_{AG}, A will learn the spectral information of clouds and various background land covers to produce more accurate attention maps.
In the proposed Auto-GAN method, T, R and A cooperate with each other and their parameters are updated together. We give the final loss function of T, R and A as follows:

$$L_{AG} = E_{c\sim d_c}[(D(F_1(c))-1)^2] + \lambda E_{c\sim d_c}[\|F_2(c)-c\|_1]$$
$$+\frac{1}{2}\{E_{f\sim d_f}[(A(f)-1)^2] + E_{x\sim d_n}[(A(x)-0)^2]\} \quad (9)$$

3 EXPERIMENTAL RESULTS

3.1 Data Description

To demonstrate the effectiveness of Auto-GAN, Sentinel-2A imagery were selected as training and testing data. Sentinel-2A is a high-resolution multi-spectral imaging satellite that carries a multi-spectral imager (MSI) for land monitoring which covers 13 spectral bands in the visible, near infrared and shortwave infrared at high spatial resolutions (10 m, 20m and 60m). The true color composite image of bands 2/3/4 with spatial resolution at 10 m was adopted as our experimental data.

The southeast of China was selected as our study area. There are many mountains, rivers and vegetation in these areas. Due to geographical factors, the economy there is more developed than other areas in China, so, there are more buildings and concrete roads in this area. Many land surfaces mentioned above are difficult to be separated from clouds, which makes it hard to detect clouds accurately. The land surface features in these images were very representative of the southeast of China. 12 Sentinel-2A Level 1C images were acquired from 1 May 2018 to 30 September 2018 from Copernicus Data Hub and cropped into 48 patches without overlapping (40 for training and 8 for testing).

3.2 Experimental Setup

For the making of training dataset, each training image patch was clipped by using a slide window with size of 256 × 256. These training patches were manually classified into two subsets: a cloudy image set and a cloud-free image set. We also extracted patches from images full of clouds to optimize A. To make full use of the image information, we rotated these patches with 90°, 180° and 270° to augment training samples. In this way, 122176 patches were obtained, where the number of cloud-free, cloudy, and all-cloud patches were 69424, 44690, and 8062 respectively. To avoid overfitting during training, the dropout rates of the attentive network, generative networks and discriminative network were set to 0.75, 1.0, 1.0 respectively. The batch size was set to 1 and the epoch was set to 4 (330000 iterations) for the training of all methods.

In the experiments, we compared Auto-GAN with three baseline methods: Sen2cor, U-net and Deeplab-v3. Sen2cor is a processor

for Sentinel-2A Level 2A product (Main-Knorn et al, 2017), and the default Sentinel-2 cloud mask. U-net firstly connects down-sampled feature maps with up-sampled feature maps to make full use of the image features in image segmentation (Ronneberger et al, 2017). Deeplab-v3 proposes atrous spatial pyramid pooling (ASPP) and has shown state-of-the-art performance in image segmentation (Chen et al, 2017).

For all methods, we adopted Adam-optimizer as the optimizer to train the networks and its parameters were fixed as: Beta1 = 0.9, Beta2 = 0.999 , the initial learning rate = 0.0002, and the exponential decay with decay rate=0.96 is used as the decay policy. Our training and validation experiments were both conducted with the TensorFlow platform on Windows 7 operation system with 16 Intel (R) Xeon CPU E5-2620 v4 @ 2.10 GHz and an NVIDIA GeForce GTX 1080Ti with 11 GB memory.
For the model training, the inputs of the proposed Auto-GAN method are images and the corresponding image-level labels (a single value to indicate whether there are clouds in the images). The inputs of baseline DL-based methods are images and the corresponding pixel-level labels (binary cloud masks).

To predict pixel-level labels of testing images, we apply the well-trained attentive network (A) on testing images with a slide window of size 256×256. Overlapping is imposed when sliding window across the testing image with a stride of 128 pixels to avoid boundary effects. Since the outputs of Auto-GAN are the attention of clouds, and the outputs of baseline methods are the probabilities of clouds, the thresholds were set respectively to 0.3 for Auto-GAN and 0.5 for baseline methods to obtain binary masks of the outputs.

To quantitatively assess the performances of the proposed Auto-GAN and baseline methods, the ground truths manually labeled are compared with binary masks, using the following measures:

$$\mathrm{OA} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FP}} \tag{10}$$

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{11}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{12}$$

$$\text{F1-score} = \frac{2\mathrm{Precison} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{13}$$

Where TP = the numbers of true positives (true cloud areas)
FP = the numbers of false positives
TN = the numbers of true negatives
FN = the numbers of false negatives

The OA represents overall accuracy of cloudy and cloud-free regions obtained by the method in the true cloud and cloud-free regions. Precision represents the accuracy of true cloud regions correctly detected by the method over all cloud regions obtained by the method. Recall represents the accuracy of cloud regions obtained by the method in true cloud regions. The F1-score is a weighted harmonic mean of Precision and Recall. The higher the values of these precision indices, the better the performance of cloud detection method.

3.3 Results Analysis

The details of visual results are shown in Figure 5. The cloud detection performance of Auto-GAN is tested over different underlying surfaces of buildings, bare land, vegetation and water.

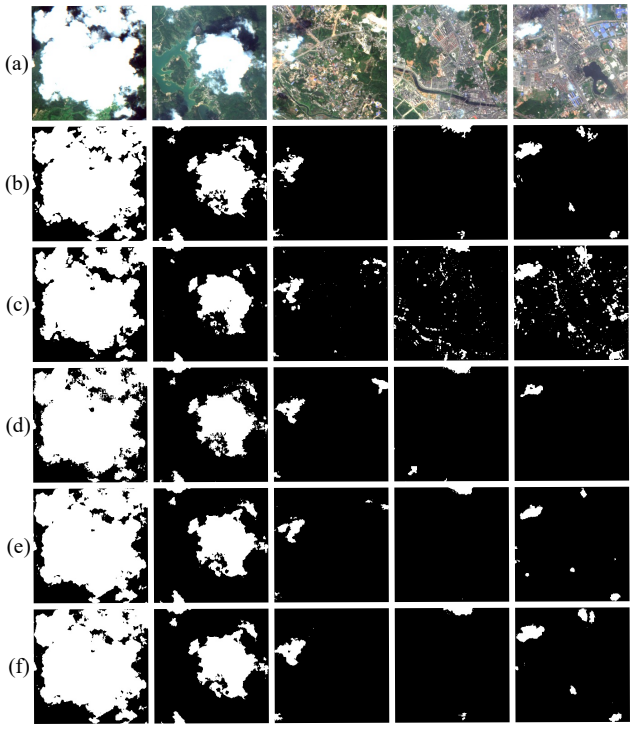

Figure 5. Visual comparisons of different cloud detection algorithms for thick clouds. (a) Original image. (b) Ground truth. (c) Sen2cor. (d) U-Net. (e) Deeplab-v3. (f) Auto-GAN.

In the visual results, white pixels represent cloud and black pixels represent background.

It can be seen that the thick clouds can easily be detected by all methods. However, the results of Sen2cor contain highlight areas and less clouds most of the time. The reason is that Sen2cor mainly uses the spectral information of the image, thus it is not sensitive to the shape of the objects and the boundary of clouds. U-Net and Deeplab-v3 can detect most of the clouds, but their results still contain few highlights, especially U-Net. All the baseline methods usually cannot distinguish well between highlights and clouds. In contrast, the proposed Auto-GAN method can always distinguish clouds from highlights, and the cloud detection results are very similar to the reference.

Methods	Sen2cor	U-Net	Deeplab-v3	Auto-GAN
OA	98.66	99.12	99.37	**99.38**
Precision	91.02	91.86	93.67	**94.13**
Recall	79.72	89.85	**93.22**	92.89
F1-score	85.00	90.85	93.45	**93.50**

Table 1. Objective evaluation for different methods.

The OA, Precision, Recall and F1-score values of all the methods are shown in Table 1. The best results have been marked in bold typeface. As presented, Auto-GAN can always obtain better performance than all baseline methods on OA, Precision and F1-score values. The recall value of Auto-GAN is only slightly lower than that of Deeplab-v3. For example, the F1-score value of Auto-GAN is 93.50% while the F1-score values of Sen2cor, U-Net and Depplab-v3 are 85.00%, 90.85% and 93.45%, respectively. It is worth noticing that the training of Auto-GAN only requires image-level labels while U-Net and Deeplab-v3 require pixel-level labels during the training of the network. While requiring much less labels in training phase, Auto-GAN outperformed all baseline methods on this Sentinel-2 dataset across all but one measure, with the only exception of a slightly lower recall measure than Deeplab-v3.

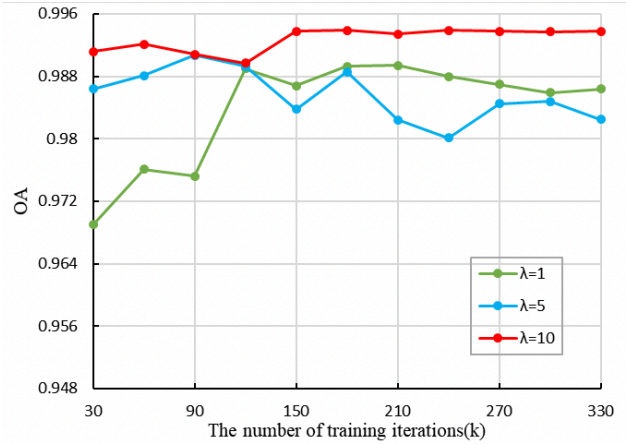

Figure 6. Sensitivity analysis for the hyper-parameter λ in the loss function of Auto-GAN on the test data.

Frequency	λ=1	λ=5	λ=10
OA	98.93	98.64	**99.38**
Precision	88.94	84.27	**93.31**
Recall	88.48	86.64	**93.67**
F1-score	88.71	85.44	**93.49**

Table 2. Objective evaluation for proposed method Auto-GAN with different values of hyper-parameter.

In the proposed method Auto-GAN, the weight λ of global cycle consistency loss is the only hyper-parameter in the L_{AG} loss function. Fig. 6 shows the comparison of overall accuracy of Auto-GAN on test data with three values of the hyper-parameter λ = 1, 5, 10 and different number of training iterations. It can be seen that in the first 120000 iterations, the greater the value of λ is, the greater the value of OA. Overall accuracy OA is higher with λ = 10 than with λ = 1 or λ = 5. The reason is that when the value of λ increases, the generative networks will be better trained, which will benefit the training of the attentive network. The values of OA, Precision, Recall and F1-score under different values of λ are shown in Table 2. It can be seen that the Auto-GAN has the best performance with λ =10.

For training, the baseline methods require images and corresponding pixel-level labels (binary masks), while Auto-GAN only uses images and corresponding image-level labels (whether the image contains clouds or not). On average, it takes more than 20 hours (1200 minutes) for a human operator to annotate pixel-level labels for a Sentinel-2A image with size 10980 × 10980 pixels. However, it only takes about 12 minutes (100 times faster) on average to annotate image-level labels for all patches cropped from the same Sentinel-2A image by the same person. So, the proposed Auto-GAN can be applied to other satellite images very quickly and efficiently.

For the translation from a cloudy image to cloud-free image, the translation results of the images containing small and large clouds are shown in Figs. 7 and 8 (in Appendix), respectively. It can be seen that the translation tunnel performs better on small clouds than large clouds. This is because with smaller cloud regions, the translation tunnel can get more information from background regions to translate the clouds into background. On the contrary, there is not enough background information in images with large cloud regions for translating the clouds into background.

4 CONCLUSION

In this work, a novel GAN-based method was proposed for cloud detection in high resolution remote sensing images. In particular, the proposed Auto-GAN framework outputs the pixel-level labels only using image-level labels in training, which reduces the time of annotating training samples by a factor of 100 compared to manually annotating pixel-level labels. Furthermore, Auto-GAN method designs three complementary tunnels to simultaneously detect cloud regions and translate images. The injecting of attention mechanism is beneficial for generating high quality images which in turn improves the performance of the attention tunnel. The Auto-GAN is trained and tested on Sentinel-2A images over China. Only the well-trained attentive network is used to predict cloud regions on the testing images. The Auto-GAN method outperforms all baseline methods on overall accuracy, precision and F1-score, and all but one method on recall measure (marginally under-performing). Both visual and quantitative analyses of experimental results demonstrate that Auto-GAN framework is very effective on cloud detection in remote sensing images.

In the future, we will focus on the following works: 1) reducing the computing time by designing simpler and more efficient networks; 2) making a dataset of bright buildings and testing the performance of our method on cloud detection on these surfaces; 3) further reducing human labor in labelling training data by training a binary classifier to annotate image-level labels of input images automatically.

REFERENCES

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2017. Re-thinking Atrous Convolution for Semantic Image Se gmentation. *arXiv Prepr*. arXiv1706.05587v3.

Fisher, A., 2014. Cloud and Cloud-Shadow Detection in SPOT5 HRG Imagery with Automated Morphological Feature Extraction. *Remote Sens*, 6(1), 776-800.

Frantz, D., Röder, A., Udelhoven, T., Schmidt, M., 2015. Enhancing the detectability of clouds and their shadows in multitemporal Dryland Landsat imagery: extending Fmask. *IEEE Geosci. Remote Sens. Lett*, 12(6), 1242–1246.

Goff, M.L., Tourneret, J.Y., Wendt, H., Ortner M., Spigai, M., 2017. Deep learning for cloud detection. *8th Int Conf. Pattern Recognit. Syst*, 1-6, Madrid.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Bing, X., Warde-Farley, D., Ozair, S., 2014. Generative adversarial nets. *Adv. Neural Inf. Process. Syst*, 27, 2672—2680.

Han, Y., Kim, B., Kim, Y., Lee, W.H., 2014. Automatic cloud detection for high spatial resolution multi-temporal images," *Remote Sens. Lett*, 5(7), 601-608.

He, K.M., Zhang, X.Y., Ren, S.P., Sun, J., 2016. Deep residual learning or image recognition. *IEEE Conf. Comput. Vis. Pattern Recognit*, Las Vegas, USA. 770-778.

Isola, P., Zhu, J., Zhou T., Efros, A. A., 2017. Image-to-Image Translation with Conditional Adversarial Networks. *2017 IEEE Conf. Comput. Vis. Pattern Recognit*., *Honolulu, HI*, 5967-5976.

Jégou, S., Drozdzal, M., Vazquez, D., Romero, A., Bengio, Y., 2017. The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation. *IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, Honolulu, HI*, 2017, 1175-1183.

Li, Z.W., Shen, H.F., Qing, C., Liu, Y.H., You, S.C., He, Z.Y., 2019. Deep learning based cloud detection for medium and high resolution remote sensing images of different sensors. *ISPRS J.*

Photogramm. Remote Sens, 150, 197-212.

Main-Knorn, M., Pflug, B., Louis, J., Debaecker, V., Müller-Wilm, U., Gascon, F., 2017. Sen2Cor for Sentinel-2. *Image and Signal Processing for Remote Sens*, 10427, Warsaw, Poland.

Mateo, G., Gonzalo G., Gomez, C., 2017. Convolutional neural networks for multispectral image cloud masking. *IEEE Int Geosci. Remote Sens*. *Symp*, 2255-2258.

Novo-Fernández, A., Franks, S., Wehenkel, C. López-Serrano, P.M., Molinier, M. and López-Sánchez, C. A. 2018. Landsat time series analysis for temperate forest cover change detection in the Sierra Madre Occidental, Durango, Mexico. *International Journal of Applied Earth Observation and Geoinformation*, 73, pp. 230-244.

Parmes, E., Rauste, Y., Molinier, M., Andersson, K. and Seitsonen, L., 2017. Automatic Cloud and Shadow Detection in Optical Satellite Imagery Without Using Thermal Bands—Application to Suomi NPP VIIRS Images over Fennoscandia. *Remote Sensing - special issue on Atmospheric Correction of Remote Sensing Data*, 9(8), pp. 806.

Qian, R., Tan, R.T., Yang, W., Su, J., Liu, J., 2018. Attentive Generative Adversarial Network for Raindrop Removal from A Single Image. *IEEE Conf. Comput. Vis. Pattern Recognit, Salt Lake City, UT,* 2482-2491.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 9351, 234-241.

Sun, L., Liu, X., Yang, Y., Chen, T. T., Wang, Q., Zhou, X., 2018. A cloud shadow detection method combined with cloud height iteration and spectral analysis for Landsat 8 oli data. ISPRS Journal of Photogrammetry and Remote Sensing. 138, 193-207.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser Ł. and Polosukhin, I., 2017. Attention Is All You Need. *Advances in Neural Information Processing Systems 30 (NeurIPS 2017).*

Wei, J., Sun, L., Jia, C., Yang, Y., 2016. Dynamic threshold cloud detection algorithms for MODIS and Landsat 8 data. *IEEE Int Geosci. Remote Sens. Symp*, 566-569.

Wang, L., Zhao, G.X., Jiang, Y.M., Zhu, X.C., Chang, C.Y., Wang, M.M., 2016. Detection of cloud shadow in Landsat 8 OLI image by shadow index and azimuth search method. J. Remote Sens. 20 (6), 1461–1469.

Xie, F., Shi, M., Shi, Z., Yin, J., Zhao, D., 2017. Multilevel cloud detection in remote sensing images based on deep learning. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 1-10.

Zhang, Y., Rossow, W.B., Lacis, A.A., Oinas, V., Mishchenko, M. I., 2004. Calculation of radiative fluxes from the surface to top of atmosphere based on ISCCP and other global data sets: Refinements of the radiative transfer model and the input data. J. Geophys. Res. 109(D19), D19105.

Zhang, Z., Iwasaki, A., Xu, G.D., Song. J.N., 2018. Small Satellite Cloud Detection Based On Deep Learning and Image Compression. *Preprint*. doi:10.20944/preprints201802.0103.v1.

Zhao, R., Ouyang, W., Li, H., Wang, X., 2015. Saliency detection by multi-context deep learning. *IEEE Conf. Comput. Vis. Pattern Recognit, Boston, MA*, 1265-1274.

Zhu, J., Park, T., Isola, P., Efros, A.A., 2017. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. *IEEE Int. Conf. Comput. Vis, Venice*, 2242-2251.

Zhu, Z., Wang, S., Woodcock, C.E., 2015. Improvement and expansion of the Fmask algorithm: cloud, cloud shadow, and snow detection for Landsats 4-7, 8, and Sentinel 2 images. Remote Sens. Environ. 159, 269–277.

Zhu, Z., Woodcock, C.E., 2012. Object-based cloud and cloud shadow detection in Landsat imagery. *Remote Sens. Environ*, 118, 83-94.

APPENDIX

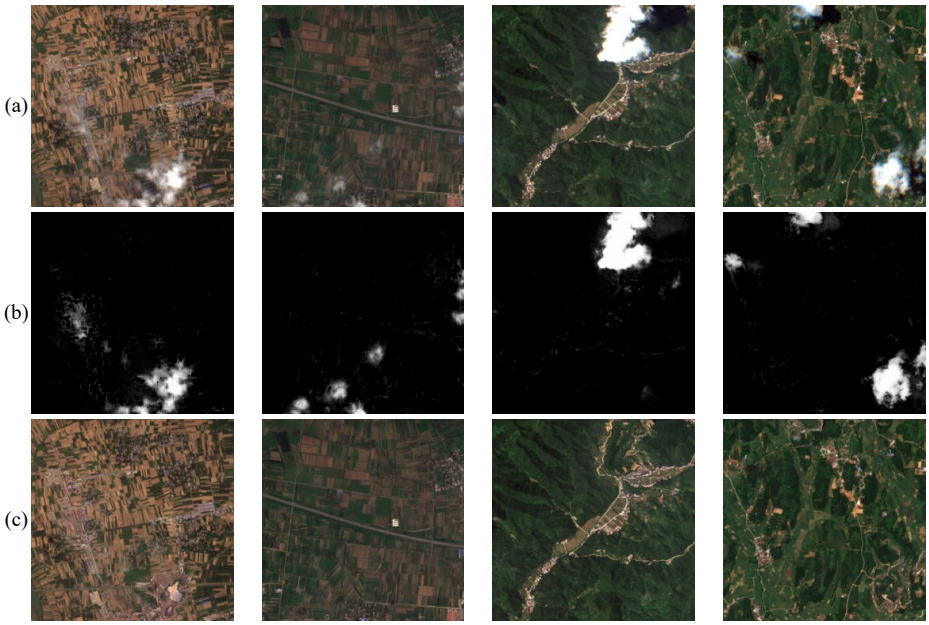

Figure 7. Successful example results of our method on Sentinel-2A image patches at 256×256 resolution. (a) Input image. (b) attention map. (c) Generated cloud-free image.

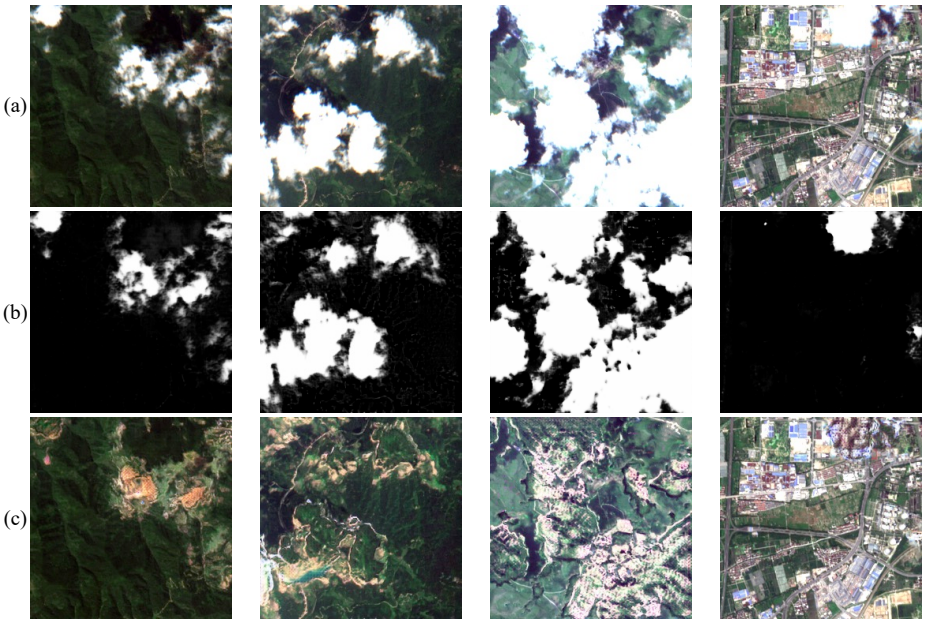

Figure 8. Failed example results of our method on Sentinel-2A image patches at 256×256 resolution. (a) Input image. (b) attention map. (c) Generated cloud-free image.